エージェントベースロボットシステム：梅

Agent-based Robot System: UME

○正 北垣 高成 , 末広 尚士 , 堀 俊夫 , 正 齋藤 史倫 (電総研)

Kosei KITAGAKI, Takashi SUEHIRO, Toshio HORI, and Fuminori SAITO
Electrotechnical Laboratory, 1-1-4 Umezono, Tsukuba, 305-8568 JAPAN

e-mail: kitagaki@etl.go.jp, suehiro@etl.go.jp, toshi@etl.go.jp, and saitoo@etl.go.jp

Abstract: To perform advanced tasks such as pliant object handling tasks by robots, components such as manipulators and sensor system should be integrated and cooperated. An easy integration of existing components helps reducing costs. This paper introduces a new robot system UME which is applied of a communication technique between agents developed for the sensor based manipulation system TAKUMI presented by the authors. The UME provides environment for developers to integrate those components with general-purpose network.

Keywords: robot system, multi-agent, system integration, object oriented, distributed system

1 はじめに

柔軟物のハンドリング等の高度な作業をロボットで行わせるには、マニピュレータやセンサシステム等のコンポーネントを有機的に結合し協調させる必要がある。ひとつの統合システムを構築する毎に、必要となる複数のコンポーネントを再構築することは非常に手間のかかることであり、すでに開発されたコンポーネントを自由に組み合わせることができればコストを低く抑えることができる。

既存のコンポーネントはソケットを介したネットワーク結合が可能となっているものが多い。シリアルポートやアナログ入出力のみのコンポーネントについては、ソケット通信をサポートしているPCなどと結合することによりネットワーク接続ができる。したがって、このネットワークを利用すれば、それぞれのコンポーネント同士が通信することは可能である。しかし、どのような情報をどのような手順で通信するかといった点に関しては開発者に任されることとなり、通信プロトコルを共通化しておかなければ、通信のためのプログラム開発に多くの時間を費やすこととなる。

分散オブジェクトを実現するための通信プロトコル標準として、近年、CORBA (Common Object Request Broker Architecture)[1] や DCOM(Distributed Common Object Model)[2] 等が提案されている。それらの上位に統合システムを構築するためのインターフェースを開発すれば汎用性が増す。さきに提案された DCOM をベースとする ORiN (Open Robot Interface for the Network)[3] は産業用ロボット用インターフェース標準として期待される。重要な点はインターフェースの定義をオープンにすることである。一方、筆者らはセンサベーストマニピュレーションシステム TAKUMI を開発し [4]、それを用いて実験を行ってきた [5]。TAKUMI はトランスピュータを用いたマルチエージェントに基づくシステムである。同システムでは、エージェントと呼ばれる機能モジュールを複数準備し、独自に定義した通信プロトコルによりエージェント間通信することで、システム全体の制御を可能としている。さらに、複数のエージェントを管理するため、通信用ルーティングパス、プロセッサ、エージェント、コマンドの定義ファイルを処理することで、システム構築に必要となるヘッダファイルやライブラリを自動生成するユーティリティが開発されている。このユーティリティを用いることでシステムの拡張が容易となっている。

本稿では、システム TAKUMI で開発されたエージェント間通信手法を適用することにより、汎用ネットワークを介してマニピュレータやカメラなどの装置をエージェントとして扱い、統合することのできるロボットシステム UME を紹介する。

2 システム概要

UME は TAKUMI のエージェントベースの考え方を継承したシステムである。TAKUMI 同様、ユーザはエージェントに対してコマンドを送ることでマニピュレータやカメラを制御したり、データを受け取ることができる。異なる点はエージェント間通信を一般的なソケットで実装したことにある。

TAKUMI においてエージェントはエージェント間通信を制御する Head とエージェント固有のデータ処理を行う Body で構成されている。トランスピュータネットワークにおける通信をサポートする上位レベルの一般的なドライバがなかったため、Head では通信データパケットのルーティング制御さえも行う必要がある。下位レベルでプログラミングできたため高速な通信が可能であるというメリットがある一方、トランスピュータ依存の形となっていた。それに対し、UME では一般的なソケット通信 (TCP/IP) を利用することでプラットフォームに依存しない形となっている。ただし、通信速度はトランスピュータネットワーク内と比較すると劣る。

3 エージェント構築手順

コンピュータの定義等一部異なるが、基本的には TAKUMI と同様の手順である。

3.1 エージェントコマンドの入出力定義記述

まず、コンピュータ、エージェント、コマンドを定義したファイルを作成する。この定義ファイルは CORBA における IDL (Interface Definition Language) にあたるものである。トランスピュータネットワークの定義ファイルを転用しているため、Occam[6] ライクな記述となっている。書式は以下の通り

である。

```
NODE <ComputerName> :                                          (1)
NETWORK                                                        (2)
  SET <ComputerName> <MachineName>                             (3)
:                                                              (4)
--C PLACE <AgentName> AT <ComputerName> <AgentAddress>  (5)
--C <CommandName> <AgentName> <Arguments> - <ReturnValues>
                                                               (6)
```

(1) ～ (4) はコンピュータの定義である。(5) はエージェントの定義である。(6) はコマンドの定義である。それぞれ複数定義可能である。このようにして記述されたファイルをユーティリティに処理させることにより、システム構築に必要となるヘッダファイルやライブラリ等は自動生成される。現在の対応言語は C と C++ であるが、加えて、euslisp 用のメソッドとオブジェクト記述プログラムも同時に生成される。また、OS としては Solaris、VxWorks、Windows をサポートしている。

3.2 エージェント Body のプログラミング

図 1 にプログラムのフレームを示す。X ウインドウのプログラミングスタイルに似た形式である。メインとなる関数で ume_server() をコールする。ume_server() は本システムが予め準備している関数であり、他のエージェントからのリクエストを受け取ったときに agent() にデータを引き渡す。agent() では引数として渡されるデータに含まれるコマンドコード(5 バイト目)に従い、その処理をユーザが記述する。本システムでは通信プロトコルとして TAKUMI における NET プロトコルをそのまま使用している (図 2)。なお、サーボなど実時間処理が必要な場合には、実時間処理の可能なプラットフォーム上で、ume_server() を呼び出す前に、タイマ割り込みを起動させておけばよい。

4 システム構成

現在構築中のシステムのコンポーネントは、双腕マニピュレータ (PA10、力覚センサ、グリッパ各 2 台)、ハンドアイシステム (P Series ＋雲台＋ステレオビジョン)、ジョイスティック 2 台 (力覚センサ利用) などである。システム構成図を図 3 に示す。実時間性を考慮すべきエージェントは VxWorks 上に実装した。PA10 はシステム既存の運動制御ボードをそのまま用いて制御された。サンプリング時間は 5ms である。なお、PA10 を動作させせるエージェントは PA ライブラリ [7] を VxWorks に移植したものを用いて開発された。ネットワーク越しにコマンドを転送しデータを受け取るための通信時間はおよそ 3ms であった。個々のコンポーネント間の通信量は数十バイト程度と少ないため、通信時間のほとんどがオーバーヘッドと考えられる。ジョイスティックにより PA10 を動作させるプログラムを起動させたところ 5ms 周期で動作することを確認した。なお、ネットワークはプライベート Lan で構成されている。

```
 1:  #if TORNADO
 2:  AgentName()
 3:  #elseif SOLARIS
 4:  int main(int argc, char **argv)
 5:  #endif
 6:  {
 7:    ume_server(A_AgentName);
 8:  }
 9:
10:  int agent(char *buf)
11:  {
12:    switch(buf[4]) {
13:    case AgentCommand1:
14:      prcesses1();
15:      break;
16:    case AgentCommand2:
17:      prcesses2();
18:      break;
19:    }
```

Fig.1　Fame of an agent program

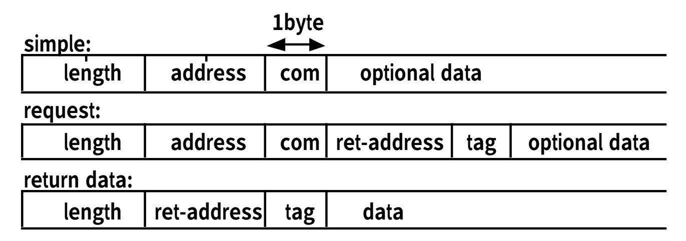


Fig.2　NET protocol definition

5 まとめ

マニピュレーションシステム TAKUMI で開発されたエージェント支援ソフトウェアを改造することで、様々なシステムコンポーネントを統合可能とするシステム UMEについて示した。

今後、システムの汎用性をより確保するため、CORBA による実装を検討する。また、本システムを用いて柔軟物ハンドリング等の実験を行う予定である。

最後に ,日頃よりご討論頂いている電子技術総合研究所ロボットグループの皆様に感謝の意を表します .

参考文献

[1] http://www.omg.org/
[2] http://www.microsoft.com/com/tech/DCOM.asp
[3] 日本ロボット工業会、"ORiN(Open Robot Interface for the Network) の概要 "
[4] 北垣高成 , 末廣尚士 , 小笠原司 , 劉雲輝 , " 並列処理型センサベーストマニピュレーションシステム : 匠 ", 日本ロボット学会誌 , Vol.15, No.3, pp.363-372, 1997.
[5] Kosei Kitagaki, Motoyoshi Fujiwara, Takashi Suehiro, and Tsukasa Ogasawara, Proc. of IEEE Int. Conf. on Intelligent Robots and Systems, Vol.2, pp.832-837, 1999.
[6] Occam2 Reference Manual, INMOS Ltd., Prentice Hall, 1988.
[7] " 可搬式汎用知能アーム取扱説明書 "、三菱重工業（株）

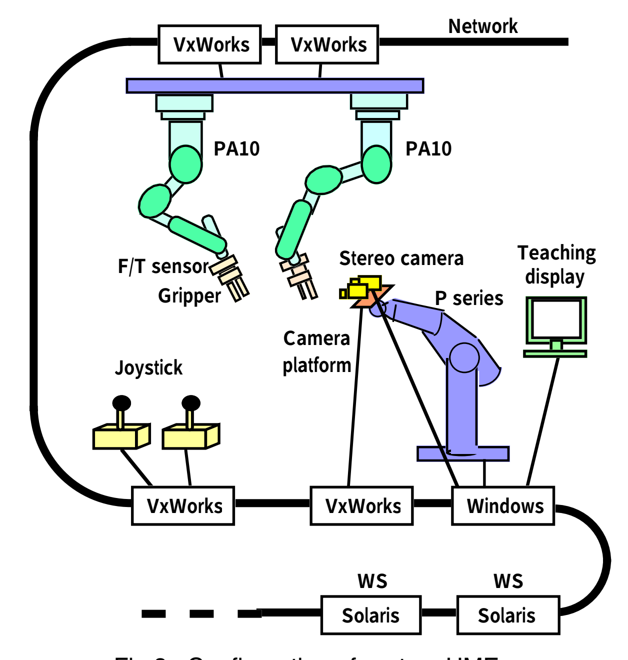


Fig.3　Configuration of system UME